Computing the Forward Reachable Set for a Multirotor Under First-Order Aerodynamic Effects

Kim, Suseong ; Falanga, Davide ; Scaramuzza, Davide

Abstract: Collision avoidance plays a crucial role in safe multirotor flight in cluttered environments. Even though a given reference trajectory is designed to be collision free, it might lead to collision due to imperfect tracking caused by external disturbances. In this work, we tackle this problem by computing the Forward Reachable Set (FRS), which is the set of positions and velocities that a multirotor can reach while following a reference trajectory due to tracking errors. Hence, if the FRS is computed before flight, we can utilize it to check the safety of a given trajectory in terms of collision avoidance. To compute a realistic FRS that covers an agile flight envelope, we consider first-order aerodynamic effects, which have the most salient influence on the vehicle. For computing FRS, we conduct a thorought stability analysis including these aerodynamic effects. Then, we present a FRS computation method which can easily be adapted to newly given reference trajectories. The presented method is validated by comparing the FRS with real flight data collected during agile and high-speed flight. In addition, we compare the FRS computed with and without compensating for firstorder aerodynamic effect to highlight their significance on the trajectory tracking performance. To the best of our knowledge, this is the first attempt to compute FRSs by incorporating firstorder aerodynamic effects for multirotors.

DOI: https://doi.org/10.1109/lra.2018.2848302

Computing The Forward Reachable Set for a Multirotor Under First-Order Aerodynamic Effects

Suseong Kim, Davide Falanga and Davide Scaramuzza

***Abstract*—Collision avoidance plays a crucial role in safe multirotor flight in cluttered environments. Even though a given reference trajectory is designed to be collision free, it might lead to collision due to imperfect tracking caused by external disturbances. In this work, we tackle this problem by computing the Forward Reachable Set (FRS), which is the set of positions and velocities that a multirotor can reach while following a reference trajectory due to tracking errors. Hence, if the FRS is computed before flight, we can utilize it to check the safety of a given trajectory in terms of collision avoidance. To compute a realistic FRS that covers an agile flight envelope, we consider first-order aerodynamic effects, which have the most salient influence on the vehicle. For computing FRS, we conduct a thorought stability analysis including these aerodynamic effects. Then, we present a FRS computation method which can easily be adapted to newly given reference trajectories. The presented method is validated by comparing the FRS with real flight data collected during agile and high-speed flight. In addition, we compare the FRS computed with and without compensating for first-order aerodynamic effect to highlight their significance on the trajectory tracking performance. To the best of our knowledge, this is the first attempt to compute FRSs by incorporating first-order aerodynamic effects for multirotors.**

***Index Terms*—Aerial systems, mechanics and control, motion control**

I. Introduction

TRAJECTORY planning and control for Micro Aerial Vehicles (MAVs) represent very active research fields in robotics. Among MAVs, recent work has shown that multirotors are capable of executing very agile maneuvers in confined spaces [1], and different solutions for computing trajectories to let a quadrotor safely fly in cluttered environments have been proposed [2], [3]. A large body of the literature relies on model-based controllers to make multirotors follow collision-free trajectories. Such controllers typically use partial or simplified dynamical models, which do not take into account aerodynamic effects [4], [5]. Hence, tracking a collision-free trajectory might still lead to collisions due to tracking errors caused by such unmodeled effects. This is especially true at high speeds, where these effects are no longer negligible.

For analyzing the tracking errors, computing Forward Reachable Sets (FRS) is a viable solution [6]. Intuitively, FRSs are the subsets of the state space that a dynamical system can reach while following a desired trajectory due to tracking errors. For the case of a multirotor, such FRSs are the sets of positions and velocities the robot could reach while flying a trajectory in space. If a reference trajectory is given before flight, we can compute the corresponding FRSs and use them to check whether the tracking errors can potentially lead to a collision with obstacles.

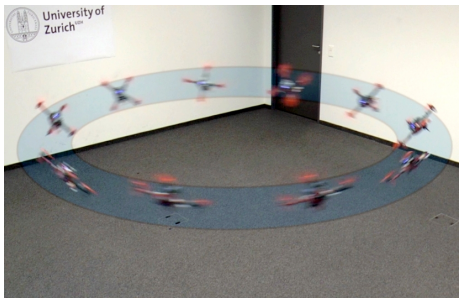


Fig. 1: A multirotor is flying in forward reachable sets.

In this work, we propose a method for *efficiently* computing FRSs for multirotors from any reference trajectories. To generate *valid* FRSs, we consider first-order aerodynamic effects as the major cause of tracking performance degradation [7]–[9]. We derive the model of the tracking error by explicitely including the aerodynamic model in the multirotor dynamic model. Then, we analyze the stability of the error dynamics to study its evolution. From the error dynamics, we show how the behavior of the errors depends on the disturbance, which can be computed from the reference velocity and acceleration, caused by the aerodynamic effects. Since external disturbances are not known a-priori, most works assume them to be bounded in norm, and conservatively use such a bound. Conversely, we compute a less conservative disturbance by exploiting aerodynamic model. Hence, the FRSs exhibit the behavior of the errors *realistically*. In addition, since the disturbance can be computed from reference velocity and acceleration, we can generate the FRSs considering discrete samples of a reference trajectory, independentely on how it is parametrized with respect to space and time. The disturbance values are then used to precompute the FRSs for each sample of the trajectory. Then, the precomputed FRSs are concatenated to obtain the overall FRSs along the entire trajectory. Since the FRSs computation only requires concatenation of precomputed FRSs, we can *efficiently* evaluate FRSs for newly designed reference trajectories.

We compare the computed FRSs with the actual errors from real flight experiments. The results show that the computed FRSs enclose the position and velocity errors well and, therefore, can be used to evaluate the safety of a reference trajectory in terms of collisions.

Manuscript received February 23, 2018; Revised May 14, 2018; Accepted June 7, 2018. This paper was recommended for publication by Editor J. Roberts upon evaluation of the Associate Editor and Reviewers' comments. This was supported by the National Centre of Competence in Research (NCCR) Robotics, the SNSF-ERC Starting Grant, and the DARPA FLA program.

The authors are with the Robotics and Perception Group, Department of Informatics, University of Zurich, and also with the Department of Neuroinformatics, University of Zurich and ETH Zurich, Switzerland—http://rpg.ifi.uzh.ch.

Digital Object Identifier (DOI): see top of this page.

A. Related Work

In this work, we aim to compute the largest deviation from a nominal trajectory a multirotor can experience due to first-order aerodynamic effects. First-order aerodynamic effects are proportional to the translational and angular velocities of the vehicle [10]. In the literature, they have been used for various purposes. For example, they have been exploited for state estimation [11]–[13]. Also, to enhance the trajectory tracking performance of a multirotor, approximated aerodynamic effects were included in the controller in [7]–[9]. Conversely, in this work, we consider the aerodynamic effects as the main source causing position and velocity errors of a multirotor, and use them to analyze trajectory tracking performance to guarantee safety.

FRSs can be used as a tool for analyzing the trajectory tracking performance of a robot [6]. For example, the trajectory planner presented in [14] utilized FRS analysis to compute the required safety margin for obstacle avoidance. However, since FRSs were computed with Hamilton-Jacobi (HJ) reachability analysis, the robot had to be controlled using dynamic programming by solving the Hamilton-Jacobi-Isaacs equation. On the contrary, our work aims to evaluate the trajectory tracking performance of a multirotor controlled with the commonly used geometric control method [4] under aerodynamic effects. In [15], the authors computed FRSs for a helicopter to design a safe trajectory (i.e., loitering motion) to perform in emergency situations. To navigate in a cluttered environment, the authors of [16] computed FRSs of predefined motion primitives. For each primitive, the corresponding FRS was optimized individually, and collision-free trajectories were planned by combining the primitives. Conversely, unlike [15] and [16], our method does not rely on precomputed reference trajectories. Also, we do not require solving a complicated optimization problem for each individual reference trajectory. Therefore, we can efficiently generate FRSs when new reference trajectories are given. Additionally, in [15] and [16], FRSs are computed using the linearized dynamic model around such a predefined trajectory, while we fully exploit the nonlinear model of a multirotor. Finally, in [14]–[16], the authors assumed the external disturbances to be bounded in their norm, and used this conservative bound to compute FRSs. On the contrary, by explicitly considering first-order aerodynamic model as the main source causing the errors, we compute FRSs in a less conservative manner.

B. Contributions

- We compute realistic FRSs for multirotors by considering a first-order aerodynamic model.
- Our computation of FRSs is very efficient because it does not require solving complex optimization problems for all reference trajectories individually.
- We compute FRSs with and without compensating the aerodynamic effects to quantitatively highlight its effect on multirotor trajectory tracking performance.
- The computed FRSs are validated with experiments including high-speed and agile flight.

C. Outline

This paper is organized as follows. First, the multirotor dynamics and the aerodynamic effect are analyzed in Sec. II-A and II-B. Then, in Sec. II-C, we introduce a multirotor controller. After that, we analyze the stability of the overall system in Sec. II-D. In Sec. III-A, we briefly present the concept of the FRS. Then, the formulation for generating FRS is described in Sec. III-B. Also, we explain how to assign the computed FRSs along a given trajectory in Sec. III-C. The experimental settings and results are summarized in Sec. IV.

D. Nomenclature

In this paper, 0_{ij} stands for the zero matrix in $\mathbb{R}^{i\times j}$, and I_i denotes the identity matrix in $\mathbb{R}^{i\times i}$. For a matrix, $\|\cdot\|$ represents the induced 2-norm, and $\lambda_{\max}(\cdot)$ and $\lambda_{\min}(\cdot)$ indicate its maximum and minimum eigenvalues. Also, $|\cdot|$ is the Euclidean norm of a vector. For two vectors $\alpha, \beta \in \mathbb{R}^3$, we denote the cross product as $\mathrm{S}(\alpha)\beta = \alpha \times \beta$. Furthermore, c and s are shorthands of $\cos$ and $\sin$, respectively.

II. Quadrotor dynamics and control

In this section, we briefly explain the multirotor dynamics and first-order aerodynamic model. Then, we present the multirotor control law taking into account the aerodynamic effect compensation. After that, we conduct the stability analysis of the multirotor error dynamics.

A. Quadrotor dynamics

To describe the multirotor dynamic model, we define the inertial $O_w\{x_w, y_w, z_w\}$ and the multirotor body-fixed $O_b\{x_b, y_b, z_b\}$ frames. We assume the body-fixed frame to be located at the center of mass of the multirotor. The translational and rotational dynamics of a multirotor are described as follows [12]:

$$\ddot{p} = -gz_w + \gamma \mathrm{z}_b - f_d \tag{1}$$

$$\dot{R} = R\mathrm{S}(\omega) \tag{2}$$

where p and R are the position and rotation matrix describing the pose of O_b with respect to O_w. The angular velocity of O_b represented in O_b is denoted as ω. The terms g and γ are the gravitational constant and the mass-normalized collective thrust, respectively. The axis z_w is defined as $e_3 = [0\ 0\ 1]^\top$, and z_b represents the axis z_b in O_w. The term $f_d \in \mathbb{R}^3$ denotes first-order aerodynamic effects which will be detailed in Sec. II-B. In (1) and (2), we consider γ and ω as the control inputs.

B. First-order aerodynamic effects

When a multirotor is maneuvering, the rotors experience both translational and rotational motion. Then, due to the multirotor motion, the rotor generates forces other than thrust. According to [7], [10], the most prominent forces acting on

the body are due to first-order aerodynamic effects, and can be expressed as:

$$\begin{aligned} f_d =& \textstyle\sum_{m=1}^{M}\{\bar{c}_1\omega_m R\pi R^\top \dot{p} + \bar{c}_1\omega_m R\pi \mathrm{S}(\omega) d_m \\ &+ \bar{c}_2\epsilon_m\omega_m R\mathrm{S}(e_3)R^\top\dot{p} + \bar{c}_2\epsilon_m\omega_m R\mathrm{S}(e_3)\mathrm{S}(\omega)d_m \\ &+ \bar{c}_3\epsilon_m\omega_m R\pi\omega - \bar{c}_4\omega_m R\mathrm{S}(e_3)\omega\}. \end{aligned} \tag{3}$$

In (3), $m = \{1, \cdots, M\}$ indicates the rotor number, $\bar{c}_{1,\cdots,4}$ are positive coefficients, ω_m is the rotor speed, ϵ_m is the rotor spinning direction ($\epsilon_{\text{odd}} = 1, \epsilon_{\text{even}} = -1$), $\pi = I_3 - e_3 e_3^\top$, and d_m is the location of the rotor in O_b. The detailed interpretation of (3) can be referred to [10] (especially, Ch. 7).

For further analysis, (3) is rewritten as :

$$\begin{aligned} f_d =& c_1 R\pi R^\top\dot{p} + c_2 R\mathrm{S}(e_3)R^\top\dot{p} \\ &+ \delta_d(\bar{c}_{1\cdots4}, \epsilon_m, \omega_m, d_m, \omega, R) \end{aligned} \tag{4}$$

where $c_1 = \bar{c}_1\sum_{m=1}^{M}\omega_m$ and $c_2 = \bar{c}_2\sum_{m=1}^{M}\epsilon_m\omega_m$ are functions depending on the rotor speeds. Also, the terms not depending on $\dot{p}$ are lumped in δ_d. Unlike c_1, which is the sum of the absolute value of rotor speeds ω_m, c_2 incorporates the rotor speed with their spinning direction $\epsilon_m\omega_m$. Since half of the rotors rotate clockwise and the others counter-clockwise, c_2 is typically much smaller than c_1. Therefore, in this work, we assume c_2 to be neglectable. For similar reasons, we further assume that the term δ_d is also negligible [11].

From the definition in (4), c_1 is a function of the rotor speeds. However, in practice, rotor speed feedback is not always available from typical off-the-shelf motor speed controllers. Hence, in [7], [8], [11], the function c_1 is approximated as a fixed parameter by assuming that a multirotor is in hover conditions. In the same manner, let us define $\hat{c}_d$ as the fixed nominal value of c_1 and $\tilde{c}_d$ as the residue such that $\tilde{c}_d = c_1 - \hat{c}_d$. The term $\tilde{c}_d$ can be caused by the rotor speed variation or wrong parameter estimation of $\hat{c}_d$. Then, from (4), first-order aerodynamic effects can be rewritten as:

$$f_d = \hat{c}_d R\pi R^\top\dot{p} + \tilde{c}_d R\pi R^\top\dot{p}. \tag{5}$$

Here, $|\tilde{c}_d|$ is bounded since its arguments, including ω and ω_m, are all bounded. Then, the final form of the dynamic equation (1) becomes:

$$\ddot{p} = -ge_3 + \gamma \mathrm{z}_b - \hat{c}_d R\pi R^\top\dot{p} - \tilde{c}_d R\pi R^\top\dot{p}. \tag{6}$$

C. Multirotor control with aerodynamic effect compensation

Let us assume that a reference trajectory for the differential flat outputs [2] $\{p^r(t), \psi^r(t)\}$ is given. The terms $p^r(t)$ and $\psi^r(t)$ are the reference position and yaw angle trajectory, respectively. Let $e_p = p - p^r$ and $e_v = \dot{p} - \dot{p}^r$ be the position and velocity errors. Then, the normalized thrust γ and desired thrust direction z_b^d can be computed as follows [7], [17] :

$$\begin{aligned} \ddot{p}^d &= F_{\text{fb}} + ge_3 + \ddot{p}^r + \hat{c}_d\dot{p} \\ \mathrm{z}_b^d &= \ddot{p}^d/|\ddot{p}^d| \\ \gamma &= (\ddot{p}^d - \hat{c}_d\dot{p})^\top\mathrm{z}_b = (|\ddot{p}^d|\mathrm{z}_b^d - \hat{c}_d\dot{p})^\top\mathrm{z}_b \end{aligned} \tag{7}$$

where $F_{\text{fb}} = -K_p e_p - K_v e_v$ with the positive diagonal gain matrice K_p and K_v.

Note 1. The control law in (7) is designed to compensate the aerodynamic effects by including $\hat{c}_d\dot{p}$ term. However, if we set $\hat{c}_d = 0$ and $\tilde{c}_d = c_1$, the control law is identical to the geometric controller [4]. In this case, we do not compensate the aerodynamic effects in the controller.

For further analysis, the dynamics in (1) are rewritten as

$$\begin{aligned} \ddot{p} &= -ge_3 + \gamma\mathrm{z}_b - \hat{c}_d R\pi R^\top\dot{p} - \tilde{c}_d R\pi R^\top\dot{p} \\ &= -ge_3 + \gamma\mathrm{z}_b - \hat{c}_d\dot{p} + \hat{c}_d(\dot{p}^\top\mathrm{z}_b)\mathrm{z}_b - \tilde{c}_d R\pi R^\top\dot{p}. \end{aligned}$$

Then, by substituting $\ddot{p}^d$, γ and z_b^d in (7), the error dynamics can be derived as follows :

$$\begin{aligned} \ddot{e}_p &= \ddot{p} - \ddot{p}^r + \ddot{p}^d - \ddot{p}^d \\ &= F_{\text{fb}} + (\gamma + \hat{c}_d\dot{p}^\top\mathrm{z}_b)\mathrm{z}_b - \ddot{p}_d - \tilde{c}_d R\pi R^\top\dot{p} \\ &= F_{\text{fb}} + |\ddot{p}^d|\{(\mathrm{z}_b^\top\mathrm{z}_b^d)\mathrm{z}_b - \mathrm{z}_b^d\} - \tilde{c}_d R\pi R^\top\dot{p} \\ &= -K_p e_p - K_v e_v + |\ddot{p}^d|\mathrm{s}_\Phi u - \tilde{c}_d R\pi R^\top\dot{p} \end{aligned} \tag{8}$$

where Φ is the angle error between the axes z_b and z_b^d, and u is the unit vector indicating the direction of $(\mathrm{z}_b^\top\mathrm{z}_b^d)\mathrm{z}_b - \mathrm{z}_b^d$ so that $\mathrm{s}_\Phi u$ is equivalent to $(\mathrm{z}_b^\top\mathrm{z}_b^d)\mathrm{z}_b - \mathrm{z}_b^d$ [4].

For the attitude dynamics in (2), we can compute the desired attitude of the multirotor R^d by having $\psi^r(t)$ and z_b^d in (7) [2]. Then, the angular rate ω can be commanded to make R follow R^d by referring to [4], and the error term Φ can be bounded. Hence, we assume that the following relation holds:

$$|\mathrm{s}_\Phi| \le \bar{\mathrm{s}}_\Phi. \tag{9}$$

The term $\bar{\mathrm{s}}_\Phi$ refers the sine value of the maximum error bound between the axes z_b and z_b^d.

D. Stability analysis

As a first step, the error dynamics of a multirotor in (8) are rewritten as:

$$\begin{aligned} \dot{e}_p &= e_v \\ \dot{e}_v &= -K_p e_p - K_v e_v + |\ddot{p}^d|\mathrm{s}_\Phi u - \tilde{c}_d R\pi R^\top\dot{p}. \end{aligned} \tag{10}$$

To analyze the stability of the error dynamics conveniently, we assume that $K_p = k_p I_3$ and $K_v = k_v I_3$, with positive scalar values k_p and k_v. Then, we define a Lyapunov candidate function as $\bar{V} = \frac{1}{2}e^\top\bar{P}e$, where $e = [e_p^\top\ e_v^\top]^\top$ and

$$\bar{P} = \begin{bmatrix} (k_p + k_v)I_3 & I_3 \\ I_3 & I_3 \end{bmatrix}.$$

The Lyapunov function is positive definite when $k_p + k_v > 1$. The directional derivative of $\bar{V}$ is:

$$\begin{aligned} \dot{\bar{V}} =& -k_p e_p^\top e_p - k_v e_v^\top e_v + e_v^\top e_v \\ &+ (e_p + e_v)^\top\{-\tilde{c}_d R\pi R^\top\dot{p} + |\ddot{p}^d|\mathrm{s}_\Phi u\}. \end{aligned} \tag{11}$$

Then, with the term $\ddot{p}^d$ in (7), $|\tilde{c}_d R\pi R^\top\dot{p}| \le |\tilde{c}_d||\dot{p}^r + e_v|$, and the assumption in (9), the above relation is rearranged further as:

$$\dot{\bar{V}} \le -\bar{e}^\top\bar{Q}\bar{e} + 2\Delta|\bar{e}| \tag{12}$$

where $\bar{e} = [|e_p| \ |e_v|]^\top$, $\Delta = |\tilde{c}_d||\dot{p}^r| + \bar{s}_\Phi|ge_3 + \hat{c}_d\dot{p}^r + \ddot{p}^r|$, and $\bar{Q} = \bar{Q}_A + \bar{s}_\Phi\bar{Q}_B + \bar{s}_\Phi\bar{Q}_C$ with

$$\bar{Q}_A = \begin{bmatrix} k_p & -\frac{1}{2}|\tilde{c}_d| \\ -\frac{1}{2}|\tilde{c}_d| & k_v - 1 - |\tilde{c}_d| \end{bmatrix}$$
$$\bar{Q}_B = \begin{bmatrix} -k_p & -\frac{1}{2}(k_p + k_v) \\ -\frac{1}{2}(k_p + k_v) & -k_v \end{bmatrix}$$
$$\bar{Q}_C = \begin{bmatrix} 0 & -\frac{1}{2}\hat{c}_d \\ -\frac{1}{2}\hat{c}_d & -\hat{c}_d \end{bmatrix}.$$

By setting $k_v > 1 + |\tilde{c}_d|$ and $k_p > \frac{|\tilde{c}_d|^2}{4(k_v - 1 - |\tilde{c}_d|)}$, we can guarantee that the matrix $\bar{Q}_A$ is positive definite. On the other hand, the matrices $\bar{Q}_B$ and $\bar{Q}_C$ cannot be positive definite since $\det(\bar{Q}_B) = -\frac{1}{4}(k_p - k_v)^2 \leq 0$ and $\det(\bar{Q}_C) = -\frac{1}{4}\hat{c}_d^2 \leq 0$. However, even though $\bar{Q}_B$ and $\bar{Q}_C$ are not positive definite, we can ensure that the matrix $\bar{Q}$ is positive definite with the condition $\bar{s}_\Phi < -\lambda_{\min}(\bar{Q}_A)/(\lambda_{\min}(\bar{Q}_B) + \lambda_{\min}(\bar{Q}_C))$. This can be easily achieved by suitably designing an attitude controller, such that $\bar{s}_\Phi \approx 0$. Therefore, with the gain values k_p and k_v fulfilling $\lambda(\bar{Q}) > 0$, (12) becomes:

$$\begin{aligned}\dot{\bar{V}} &\leq -\lambda_{\min}(\bar{Q})|\bar{e}|^2 + 2\Delta|\bar{e}| \\ &\leq -\lambda_{\min}(\bar{Q})(1-\theta)|\bar{e}|^2 - \lambda_{\min}(\bar{Q})\theta|\bar{e}|^2 + 2\Delta|\bar{e}|\end{aligned}$$

where $0 < \theta < 1$. Finally, we have:

$$\dot{\bar{V}} \leq -\lambda_{\min}(\bar{Q})(1-\theta)|\bar{e}|^2, \text{ for } |\bar{e}| \geq \frac{2\Delta}{\lambda_{\min}(\bar{Q})\theta}. \quad (13)$$

Therefore, the error dynamics of the translational system of a multirotor is uniformly ultimately bounded [18].

Note 2. In this analysis, we concluded that $|\bar{e}|$ will eventually enter into the ultimate bound, and the size of the ultimate bound is proportional to Δ which is

$$\Delta = |\tilde{c}_d||\dot{p}^r| + \bar{s}_\Phi|ge_3 + \hat{c}_d\dot{p}^r + \ddot{p}^r|. \quad (14)$$

The terms on the right-hand side can be identified $\{\hat{c}_d, |\tilde{c}_d|, \ \bar{s}_\Phi\}$ or directly available $\{\dot{p}^r, \ \ddot{p}^r\}$ so that we can compute Δ before flight.

III. Computing Forward Reachable Set

In Sec. II, we studied the behavior of the error dynamic system via Lyapunov stability analysis. However, from this analysis, it is difficult to quantitatively know how the error would evolve and get into the ultimate bound region. Therefore, in this section, we numerically evaluate the evolution of the error. First, we introduce the concept of Forward Reachable Set (FRS). Then, we explain how to compute the FRS with a fixed disturbance Δ in (14). After that, we describe how to concatenate them to find a FRS along with Δ which is time varying.

A. Background

In this work, the FRS refers to the possible set of position and velocity errors that a multirotor can experience while tracking a reference trajectory. Let $F(t) \subset \mathbb{R}^6$ represent the FRS of the error $e(t) \in \mathbb{R}^6$ at time $t \in [0, t_f]$ given the initial set of error $e(0) \in \xi_0$ and the closed-loop dynamics in (10). In other words, the error $e(0)$ in the set ξ_0 will evolve inside of the set $F(t)$ for $t \in [0, t_f]$ as:

$$e(0) \in \xi_0 \subset F(0) \ \Rightarrow \ e(t) \in F(t), \ \ t \in [0, t_f]. \quad (15)$$

This condition can be enforced by having two constraints. The first one is related to the evolution of the error. Let $P(t) \in \mathbb{R}^{6\times6}$ be a time varying positive definite matrix, $V(t,e) = e(t)^\top P(t) e(t)$ a Lyapunov candidate function, and $\alpha(t)$ be a positive scalar variable indicating the level of $V(t,e)$. Then, the FRS can be represented with $P(t)$ and $\alpha(t)$ as :

$$F(t) = \{e(t) | V(t,e) \leq \alpha(t)\} \quad (16)$$

with the V and α satisfying the following constraints :

$$\begin{aligned}&\dot{V}(t,\hat{e}) < \dot{\alpha}(t) \\ &\text{for} \quad \hat{e}(t) = \{e(t) | V(t,e) = \alpha(t), \ t \in [0, t_f]\}.\end{aligned} \quad (17)$$

From (17), the state $e(t) \in F(t)$ cannot escape from the sublevel set described by $V(t,e) = \alpha(t)$. Therefore, the FRS can be described with $P(t)$ and $\alpha(t)$.

The second constraint is related to the initial set of errors. The initial set ξ_0 should be a subset of $F(0)$. To satisfy the constraint, we define a positive definite matrix R_0 satisfying $\xi_0 = \{e | e^\top R_0 e \leq 1\}$. Then, the constraint can be written as follows:

$$V(0,\hat{e}) \leq \alpha(0) \ \text{ for } \ \hat{e} = \{e | e^\top R_0 e \leq 1\}. \quad (18)$$

B. Computing FRS with a fixed Δ

Since we are interested in computing the FRS having the smallest volume as possible, we can formulate a problem which optimizes $P(t)$ and $\alpha(t)$ that minimize the volume of $F(t)$ for $t \in [0, t_f]$ while satisfying (17) and (18).

To compute the FRS in (16), we define the Lyapunov function such as $V = e(t)^\top P(t) e(t)$ with the time varying positive definite matrix:

$$P(t) = \begin{bmatrix} P_p(t) & P_{pv}(t) \\ P_{pv}(t) & P_v(t) \end{bmatrix} \quad (19)$$

where $P_p(t), P_{pv}(t), P_v(t) \in \mathbb{R}^{3\times3}$ are diagonal matrices. By substituting (10) in the Lyapunov candidate, its directional time derivative can be rearranged as follows:

$$\dot{V} = -e^\top Q e + 2e^\top P B(|\ddot{p}^d| s_\Phi u - \tilde{c}_d R\pi R^\top \dot{p}) + e^\top \dot{P} e \quad (20)$$

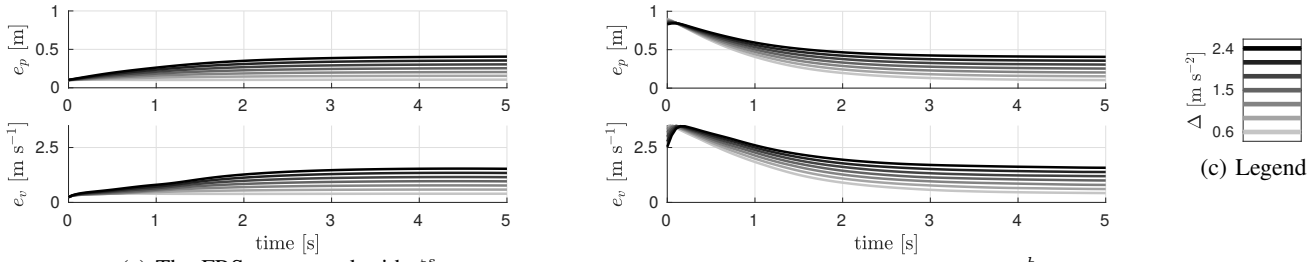
where $Q = -(A^\top P + PA)$, and

$$A = \begin{bmatrix} 0_{33} & I_3 \\ -K_p & -K_v \end{bmatrix}, \quad B = \begin{bmatrix} 0_{33} \\ I_3 \end{bmatrix}.$$

On the right hand side of (20), the second term is further developed as:

$$\begin{aligned}&e^\top P B(|\ddot{p}^d| s_\Phi u - \tilde{c}_d R\pi R^\top \dot{p}) \\ &\leq (p_{pv}|e_p| + p_v|e_v|)(\eta_p|e_p| + \eta_v|e_v| + \Delta)\end{aligned}$$

where p_{pv}, p_v, k_p, and k_v are the maximum eigenvalues of P_{pv}, P_v, K_p, and K_v, respectively. Also, $\eta_p = \bar{s}_\Phi k_p$ and $\eta_v = \bar{s}_\Phi(k_v + \hat{c}_d) + |\tilde{c}_d|$. The term Δ is from (14).



(a) The FRSs computed with ξ_0^s. (b) The FRSs computed with ξ_0^b. (c) Legend

Fig. 2: The FRSs computed with various values of Δ and ξ_0. For the coefficients, we set $\hat{c}_d = 0.0$ and $|\tilde{c}_d| = 0.35$. The other settings are available in Tab. I. Since we use same gain values for x, y, and z axes, the results are the same for all the axes.

We define two variables $\bar{e}_p, \bar{e}_v \in \mathbb{R}$ satisfying the following constraints:

$$\bar{e}_p^2 = |e_p|^2, \quad \bar{e}_p \geq 0, \quad \bar{e}_v^2 = |e_v|^2, \quad \bar{e}_v \geq 0.$$

Then, $\dot{V}$ can be further developed as follows:

$$\begin{aligned} \dot{V} \leq & -e^\top Q e + e^\top \dot{P} e \\ & + 2(p_{pv}\bar{e}_p + p_v \bar{e}_v)(\eta_p \bar{e}_p + \eta_v \bar{e}_v + \Delta). \end{aligned} \tag{21}$$

Also, we introduce the positive definite matrix $S(t) \in \mathbb{R}^{6\times 6}$ to define an ellipsoid enclosing the set of error as:

$$\hat{e}^\top(t) S(t) \hat{e}(t) \leq 1 \text{ for } \hat{e}(t) = \{e(t) | V(t,e) = \alpha(t)\}. \tag{22}$$

Since the ellipsoid is constrained to enclose the set of errors, we can minimize the volume of the FRS by minimizing the volume of the ellipsoid. The volume of an ellipsoid is monotonic with respect to $-\det(S(t))$.

Hereinafter, we describe the optimization problem to compute $P(t)$ and $\alpha(t)$ minimizing the volume of $F(t)$ using the Lyapunov analysis in (21) with the constraints in (17), (18), and (22). To deal with the problem more conveniently, we discretize the variables P, α, S, p_{pv}, and p_v in time t. The discretized time is denoted as t_n ($n = \{0, \cdots, N\}$). Then, we set an optimization problem such as :

$$\inf_{P(t_n),\alpha(t_n),S(t_n),p_{pv}(t_n),p_v(t_n)} \quad -\sum_{n=0}^{N} \det(S(t_n)) \tag{23}$$

$$\begin{aligned} \text{s.t.} \quad & \dot{\alpha}(t_n) - \dot{V}(t_n) \geq 0 \text{ for } e, \bar{e}_p, \bar{e}_v \text{ satisfying } \mathrm{c}_1 \text{ to } \mathrm{c}_9, \\ & 1 - e^\top S(t_n) e \geq 0 \text{ for } e \text{ satisfying } \mathrm{c}_9 \text{ and } \mathrm{c}_{10}, \\ & \alpha(t_0) - V(t_0) \geq 0 \text{ for } e \text{ satisfying } \mathrm{c}_{11}. \end{aligned}$$

where

$$\begin{aligned} \mathrm{c}_1 &: \ \bar{e}_p^2 = e_p^T e_p, & \mathrm{c}_2 &: \ \bar{e}_p \geq 0 \\ \mathrm{c}_3 &: \ \bar{e}_v^2 = e_v^T e_v, & \mathrm{c}_4 &: \ \bar{e}_v \geq 0 \\ \mathrm{c}_5 &: \ p_{pv}(t_n) I_3 \geq P_{pv}(t_n), & \mathrm{c}_6 &: \ p_v(t_n) I_3 \geq P_v(t_n) \\ \mathrm{c}_7 &: \ p_{pv}(t_n) I_3 \geq -P_{pv}(t_n), & \mathrm{c}_8 &: \ p_v(t_n) I_3 \geq -P_v(t_n) \\ \mathrm{c}_9 &: \ \alpha(t_n) - V(t_n) = 0, & \mathrm{c}_{10} &: \ S(t_n) > 0 \\ \mathrm{c}_{11} &: \ 1 - e^\top R_0 e = 0. \end{aligned}$$

The constraints c_5 to c_8 are added to ensure that $p_{pv}(t_n) \geq \|P_{pv}(t_n)\|$ and $p_v(t_n) \geq \|P_v(t_n)\|$. The optimization problem (23) can be solved by referring to [16].

In the optimization problem settings, the input terms can be categorized as: (i) multirotor system parameters $\{\hat{c}_d, |\tilde{c}_d|, \bar{\mathrm{s}}_\Phi\}$; (ii) controller gain values $\{K_p, K_v\}$; (iii) user selected parameters $\{\xi_0, \Delta\}$. The parameters in (i) and (ii) should be set to match with multirotor properties. On the other hand, the parameters in (iii) could vary in accordance with the initial condition or the reference trajectory. Hence, we compute FRSs with various values of ξ_0 and Δ. Let Δ_l ($l = \{1, \cdots, L\}$) be the disturbance value which can be selected to range the disturbance values of interest. For the initial condition ξ_0, we use two different settings to see how the set of error evolves to ultimate bounds from both bigger and smaller regions. The two initial conditions will be denoted as ξ_0^j with $j = \{b, s\}$ where b and s stand for *bigger* and *smaller* regions, respectively.

For each combination of input terms Δ_l and ξ_0^j, we will denote the output of the optimization as $F_l^j = \{F_l^j(t_0), \cdots, F_l^j(t_N)\}$ with the element $F_l^j(t_n) = \{e | e^\top P_l^j(t_n) e \leq \alpha_l^j(t_n)\}$ by utilizing the sub and superscripts used in the input terms. Then, the outputs, which are computed with various settings of the disturbances and initial condition values, can be quoted with l, j, and n where n indicates the sequence number of t_n.

We generated FRSs with the settings in Tab. I using MOSEK with YALMIP [19]. The computed FRSs are displayed in Fig. 2. The FRSs are projected to each coordinate for visualization purposes. There are two notable characteristics from the FRSs shown in Fig. 2. First, the final values that each FRS converged to are proportional to the value of Δ_l. Second, if the disturbance Δ_l is set to the same value, the FRSs converges to the similar values regardless of the initial condition j. These are the expected results from the analysis in (13) and Note 2.

Note 3. From Fig. 2, it is obvious that the FRSs, computed with the setting in Tab. I, are fully converged at about 5 s. The fully converged FRSs could be considered as the ultimate bound mentioned in Note 2.

param	value	param	value
K_p	$10 I_3$ [s^{-2}]	dt	0.05 [s]
K_v	$6 I_3$ [s^{-1}]	R_0^s (ξ_0^s)	blockdiag($100 I_3, 25.0 I_3$)
$\bar{\mathrm{s}}_\Phi$	0.052 [·] $\approx \sin 3°$	R_0^b (ξ_0^b)	blockdiag($1.56 I_3, 0.39 I_3$) [m^{-2}, m^{-2}s^2]
$\hat{c}_d$	(w) 0.3 [s^{-1}] (wo) 0.0 [s^{-1}]	$\lvert\tilde{c}_d\rvert$	(w) 0.05 [s^{-1}] (wo) 0.35 [s^{-1}]

TABLE I: Parameters used for computing FRSs. The (w) and (wo) represent that the parameters are chosen with and without the aerodynamic compensation, respectively.

Algorithm 1 Assigning FRSs along a reference trajectory

1: **function** CONCATENATE FRS
2: **inputs** $\Delta(t)$ in (24), ξ_0, Δ_l, and
3: $F_l = \{F_l^b, F_l^s\}$ $(l = \{1, \cdots, L\})$
4: **find** Δ_l s.t. $\Delta_{l-1} < \hat{\Delta} \leq \Delta_l$
5: where $\hat{\Delta} = \max(\Delta(t))$, $\forall t \in [t_0, t_1]$
6: $l_0 \leftarrow l$
7: **find** $F_{l_0}^{j^*}(n^*) = \text{argmin}_{F_{l_0}^j(n)} \text{Volume}(F_{l_0}^j(n))$
8: s.t. $F_{l_0}^j(n) \in F_{l_0}$, $\xi_0 \subset F_{l_0}^j(n)$
9: $n_0 \leftarrow n^*$, $j_0 \leftarrow j^*$
10: **for** $q \leftarrow 1$ to $Q-1$ **do**
11: **find** Δ_l s.t. $\Delta_{l-1} < \hat{\Delta} \leq \Delta_l$
12: where $\hat{\Delta} = \max(\Delta(t))$, $\forall t \in [t_q, t_{q+1}]$
13: $l_q \leftarrow l$
14: **find** $F_{l_q}^{j^*}(n^*) = \text{argmin}_{F_{l_q}^j(n)} \text{Volume}(F_{l_q}^j(n))$
15: s.t. $F_{l_q}^j(n) \in F_{l_q}$, $F_{l_{q-1}}^{j_{q-1}}(n_{q-1}+1) \subset F_{l_q}^j(n)$
16: $n_q \leftarrow n^*$, $j_q \leftarrow j^*$
17: **end for**
18: **return** l_q, j_q, n_q for $q = \{0, \cdots, Q-1\}$
19: **end function**

C. Concatenating FRS with respect to time varying $\Delta(t)$

In Sec. III-B, we described a method to generate FRS with a fixed disturbance value and an initial set of error. However, the disturbance $\Delta(t)$ will keep varying with respect to time in most of flight operation scenarios. Let us assume that we use only one FRS computed with Δ_l satistying $\Delta_{l-1} \leq \max(\Delta(t)) \leq \Delta_l$ for $t \in [0, t_f]$ to analyze the FRS of a given reference trajectory. In this case, the computed FRS would indicate that the set of errors will eventually converge to the ultimate bound computed with Δ_l. Then, the FRS will be too conservative since the size of the ultimate bound is proportional to the maximum norm of disturbance during the entire flight duration.

Indeed, the FRS computed in the previous section describes that the set of error $\hat{e}(t_n) = \{e | e^\top P_l^j(n) e \leq \alpha_l^j(n)\}$ will evolve to $\hat{e}(t_{n+1}) = \{e | e^\top P_l^j(n+1) e \leq \alpha_l^j(n+1)\}$ during the time interval $\mathrm{d}t$ under the disturbance Δ_l. If the overall flight time is descritized as t_q $(q = \{0, \cdots, Q\})$ using $\mathrm{d}t = t_{n+1} - t_n$ in (23), we can assign a disturbance value Δ_l for every time interval between t_q and t_{q+1} using (14) as

$$\Delta(t) = |\tilde{c}_d||\dot{p}^r(t)| + \bar{\mathrm{s}}_\Phi |g e_3 + \hat{c}_d \dot{p}^r(t) + \ddot{p}^r(t)|. \quad (24)$$

Then, by concatenating the elements of FRSs along with the Δ_l assigned to every time interval $\mathrm{d}t$, it is possible to make FRSs more tight than using a single Δ_l. Therefore, our approach for generating FRS is assigning n_q, j_q, and l_q for all $q = \{0, \cdots, Q-1\}$ to quote element $F_{l_q}^{j_q}(n_q)$ along with the time varying $\Delta(t)$.

We explain how to concatenate the FRS elements $F_l^j(n)$ along with Alg. 1. First, we compute $\Delta(t)$ for a given reference trajectory with (24). Also, we prepare F_l optimized with various Δ_l values ranging $\Delta(t)$. Furthermore, we guess the set of error ξ_0 that includes possible initial position and velocity errors (lines 2-3).

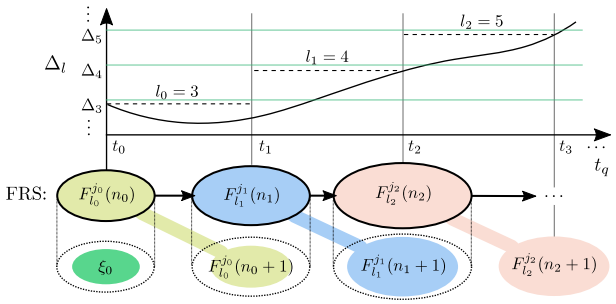


Fig. 3: Concatenating FRSs with respect to $\Delta(t)$. In the upper figure, we select l_q for each time interval. Then, in the lower figure, we select the smallest $F_{l_q}^{j_q}(n_q)$ s.t. $F_{l_q}^{j_q}(n_q) \in F_{l_q}$ and $F_{l_{q-1}}^{j_{q-1}}(n_{q-1}+1) \subset F_{l_q}^{j_q}(n_q)$.

We start concatenating the FRS from $q = 0$. First, we find l_0 indicating the disturbance for the time interval $t \in [t_0, t_1]$ by following the line 4 to 6. Then, among F_{l_0}, we find the element which can include ξ_0 as shown in line 7 to 9. For n_0 and j_0, it is possible to assign any combination of n and j indicating the elements of $\hat{F}_{l_0} = \{F_{l_0}^j(n) | \xi_0 \subset F_{l_0}^j(n), F_{l_0}^j(n) \in F_{l_0}\}$. However, since we are interested in finding FRS as tight as possible, we choose n_0 and j_0 indicating the element of $\hat{F}_{l_0}$ having smallest volume.

After that, we concatenate FRSs for $q = \{1, \cdots, Q-1\}$ by repeating the following procedure. For $t \in [t_q, t_{q+1}]$, we find l_q referring to line 11 to 13. Next, we search for the n_q and j_q as explained in line 14 to 16. Here, the $F_{l_{q-1}}^{j_{q-1}}(n_{q-1}+1)$ is the replacement of ξ_0 appeared at line 8. This is because the set $F_{l_{q-1}}^{j_{q-1}}(n_{q-1})$ at t_{q-1} evolves to $F_{l_{q-1}}^{j_{q-1}}(n_{q-1}+1)$ at t_q. Once the loop is done, we have the tight FRS for the given reference trajectory by referring l_q, j_q, and n_q for $q = \{0, \cdots, Q-1\}$. The concatenating procedures are illustrated in Fig. 3.

Note 4. To clearly deliver the proposed FRS computation method, we summarize the overall procedure as follows:

1) Collect input values for solving (23).
2) Compute $F_l = \{F_l^b, F_l^s\}$ for a range of Δ_l values.
3) Compute $\Delta(t)$ by using (24) for a reference trajectory.
4) Concatenate the elements of F_l using Alg. 1.

Once new reference trajectories are designed, we can efficiently generate FRSs for them by computing $\Delta(t)$ and concatenating the precomputed F_l.

IV. EXPERIMENTS

We conducted a number of experiments to validate the proposed FRS computation method. The detailed experimental setup, scenarios, and results are presented in this section.

A. Experimental setup

Our experimental platform is based on the DJI F330 frame. We used Cobra CM2208 motors with 6045 propellers controlled by DYS XSD30A ESCs. The desired thrust and the rate commands of the multirotor are computed by a ground control computer. The multirotor states are estimated using an OptiTrack motion-capture system, and the estimates are used to compute the commands. The commands are sent to an

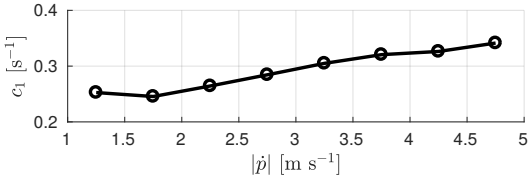


Fig. 4: The identifcation result for $c_1 = \hat{c}_d + \tilde{c}_d$.

onboard computer (UP-board), which transmits the signal to a Lumenier F4 AIO flight controller responsible for bodyrates control. The detailed software and hardware informations are public at https://github.com/uzh-rpg/rpg_quadrotor_control.

In the FRS computation, we used Δ_l for every $0.1\,\mathrm{m\,s^{-2}}$ from 0.5 to $2.5\,\mathrm{m\,s^{-2}}$. The rest of settings are in Tab. I.

B. Parameter identification for computing FRSs

We run a system identification based on [13] to find $\hat{c}_d$, $|\tilde{c}_d|$, and $\bar{\mathrm{s}}_\Phi$. We flew the multirotor along circular trajectories while varying the reference speed from 1.0 to $5.0\,\mathrm{m\,s^{-1}}$. For each $0.5\,\mathrm{m\,s^{-1}}$ interval, we collected data and estimated c_1 in (4). Fig. 4 shows that c_1 tends to become larger with respect to the flight speed. It is because the rotors speed increases with the multirotor speed in the circular trajectory. Since the parameter varies approximately from $0.25\,\mathrm{s^{-1}}$ to $0.35\,\mathrm{s^{-1}}$, we set the coefficients $\hat{c}_d$ and $|\tilde{c}_d|$ as in Tab. I. Also, during the entire flight, the maximum error between z_b and z_b^d in (7) is observed as $3\,\mathrm{deg}$. Therefore, we set $\bar{\mathrm{s}}_\Phi = \sin 3°$.

C. Experimental scenario

We designed two different experimental scenarios to conduct a quantitative analysis on the computed FRS. The first one is to show the quality of the ultimate bound, which is mentioned in Note 3, computed from FRS. The second one is to evaluate the quality of the concatenated FRSs computed by referring to Alg. 1. In both experiments, we compute FRSs with and without aerodynamic effects compensation (Note 1), and compare them with the measured tracking errors.

Scenario 1 : The goal of the first experimental scenario is to compare the measured position error with the ultimate bound. To achieve the goal, we make the multirotor fly along a circular trajectory while ramping up the speed. The diameter of the circle was set to $5\,\mathrm{m}$ and the speed was gradually increased up to $5\,\mathrm{m\,s^{-1}}$ during $50\,\mathrm{s}$. Since we increase the velocity very slowly, the predicted disturbance from (24) also changes slowly so that we can assume the multirotor to be under quasi-static disturbance. According to the definition of the ultimate bound, if the error started evolving inside of the ultimate bound, it cannot escape from it. Therefore, we start the position and velocity error of the multirotor from very small region and see whether the error stays inside of the ultimate bound during the entire flight.

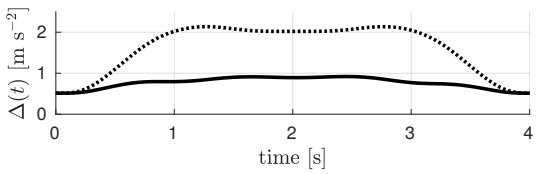


Fig. 5: The disturbance computed using (24). The solid and dotted lines represent $\Delta(t)$ histories with and without the aerodynamic compensation, respectively.

Scenario 2 : To show the quality of the FRS computed by concatenating its elements, we generate a reference trajectory that rapidly alters $\Delta(t)$ in (24) with respect to time. The reference trajectory is generated as in Fig. 7 to minimize the snap of the trajectory [3] while connecting a series of waypoints $(0,0)$, $(3,1.5)$, $(4.5,0)$, $(3,\text{-}1.5)$, and $(0,0)$. Here, the overall flight duration is set to $4\,\mathrm{s}$. Also, the maximum and average speed of the trajectory are about $4.5\,\mathrm{m\,s^{-1}}$ and $3.2\,\mathrm{m\,s^{-1}}$, respectively. The computed disturbance profiles with and without the aerodynamic compensation are described in Fig. 5. To report concrete experimental results, we fly the multirotor fifty times along the reference trajectory.

D. Experimental results and discussion

Scenario 1 : In Fig. 6, the ultimate bounds and the position errors are described. The ultimate bound for continuously varying $\Delta(t)$ is generated by interpolating the fully converged set of FRS computed with various Δ_l values as mentioned in Note 3.

Here, there are two notable aspects. First, when the reference speed is small ($|\dot{p}^r| \leq 1\,\mathrm{m\,s^{-1}}$), the position errors are comparably small in both cases. However, as we increase the reference speed, the position errors rapidly become larger if the aerodynamic effects are not compensated. This is not surprising, since the aerodynamic effects linearly depend on the velocity of the multirotor. Second, the position errors do not escape from the computed ultimate bounds. As we speed up, the position errors increase together with the size of the ultimate bound in both cases. From this tendency, it is possible to say that the computed FRSs well encircle the position errors.

Scenario 2 : In Fig. 7, we show FRSs located along the reference trajectory. Also, the position trajectories measured during fifty runs are represented. From the figure, it is clearly shown that the measured trajectories always evolved inside of the computed FRSs in both conditions. The detailed comparison between the tracking error and the FRSs are shown in Fig. 8. In the figure, together with the evolution of the errors, we can

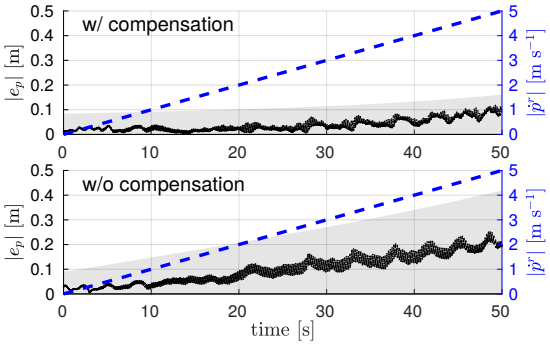


Fig. 6: The FRSs are described with the grey shaded region. The measured errors are represented with the black dots. The velocity profile is shown with the blue-dotted line.

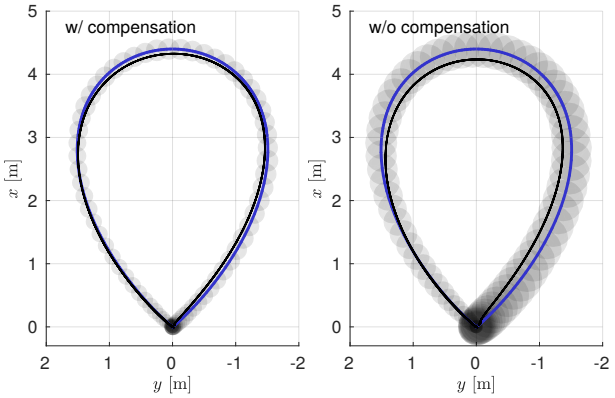


Fig. 7: The FRSs (shaded ellipsoid) are located for every $0.05\,\mathrm{s}$ around the reference trajecotry (blue line). The measured positions from fifty runs are described with black lines.

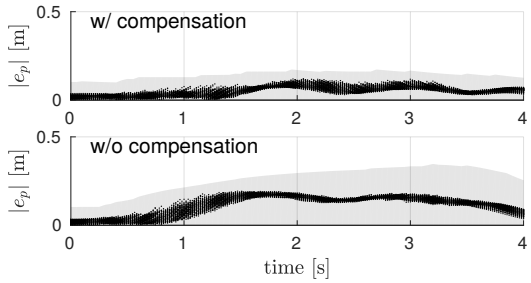


Fig. 8: The FRSs (grey shaded region) and position error histories of the fifty trials (black dots) are described.

see the evolution of the FRSs concatenated with respect to the disturbance described in Fig. 5. In both cases, the volume of the FRSs becomes larger along with the disturbance (shown in Fig. 5) until $3\,\mathrm{s}$. After that, the volume starts shrinking as like the disturbance. If we do not perform concatenation, the FRS will increase and converge to the ultimate bound computed with the maximum disturbance $|\Delta(t)|$. Therfore, the computed FRSs will be too conservative. However, by concatenating the FRSs, we can make the volume of FRSs follow the tendency of the disturbance while surrounding errors.

In both scenarios, the position errors were much bigger if we did not perform aerodynamic effects compensation, especially during high speed and agile flight. By comparing the FRSs and the errors measured with two different controllers, we can say that compensating aerodynamic effects should be considered to design a high performance trajectory tracking controller for multirotors. Furthermore, in Figs. 6 and 8, the trend of the errors matches with the evolution of the FRSs which are computed with the disturbance caused by first-order aerodynamic effects. By exploiting the known, but overlooked, aerodynamic model, we could compute less conservative FRSs than using a heuristically determined constant disturbance. From this consideration, we can conclude that the proposed FRS computation method for multirotors can be practically utilized in trajectory generation process for guaranting safe navigation in the proximity of clutter.

V. Conclusions

In this work, we proposed a method to generate forward reachable sets for multirotors in order to utilize them as a tool for verifying safety of given reference trajectories. Our study takes into account that first-order aerodynamic effects are not negligible when a multirotor is required to fly at high-speed. Using our approach, it is possible to predict the consequences of such effects on the closed-loop behaviour of a multirotor, and verify whether there might be collisions or, on the contrary, whether the trajectory is safe. Since planning obstacle-free trajectories for a robot does not guarantee that it will not experience collisions, this study provides crucial results to guarantee obstacle avoidance in cluttered environments.

References

[1] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017.

[2] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2011, pp. 2520–2525.

[3] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Proc. Int. Symp. Robot. Research (ISRR)*, 2013, pp. 1–8.

[4] T. Lee, M. Leoky, and N. McClamroch, "Geometric tracking control of a quadrotor uav on se(3)," in *IEEE Conf. Decision Control (CDC)*, Dec. 2010, pp. 5420–5425.

[5] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *Int. J. Robot. Research*, vol. 31, no. 5, pp. 664–674, 2012.

[6] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, Nov. 1999.

[7] J.-M. Kai, G. Allibert, M.-D. Hua, and T. Hamel, "Nonlinear feedback control of quadrotors exploiting first-order drag effects," in *IFAC World Congress*, vol. 50, no. 1, Jul. 2017, pp. 8189–8195.

[8] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 620–626, Apr. 2018.

[9] J. Svacha, K. Mohta, and V. Kumar, "Improving quadrotor trajectory tracking by compensating for aerodynamic effects," in *IEEE Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, Jun. 2017, pp. 860–866.

[10] R. W. Prouty, *Helicopter performance, stability, and control*, 1995.

[11] P. Martin and E. Salaün, "The true role of accelerometer feedback in quadrotor control," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2010, pp. 1623–1629.

[12] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 20–32, 2012.

[13] M. Burri, M. Dätwiler, M. W. Achtelik, and R. Siegwart, "Robust state estimation for micro aerial vehicles based on system dynamics," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2015, pp. 5278–5283.

[14] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin, "FaSTrack: A modular framework for fast and guaranteed safe motion planning," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Dec. 2017.

[15] D. Althoff, M. Althoff, and S. Scherer, "Online safety verification of trajectories for unmanned flight with offline computed robust invariant sets," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Sep. 2015.

[16] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *Int. J. Robot. Research*, vol. 36, no. 8, pp. 947–982, Jun. 2017.

[17] S. Omari, M.-D. Hua, G. Ducard, and T. Hamel, "Nonlinear control of VTOL UAVs incorporating flapping dynamics," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Nov. 2013, pp. 2419–2425.

[18] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ, USA: Prentice Hall, 1996.

[19] J. Lofberg, "YALMIP : a toolbox for modeling and optimization in MATLAB," in *2004 IEEE Int. Symp. on Computer Aided Control Systems Design*.